

**technical**

Supporting Enterprise Networks and Operating Environments

# SUPPORT

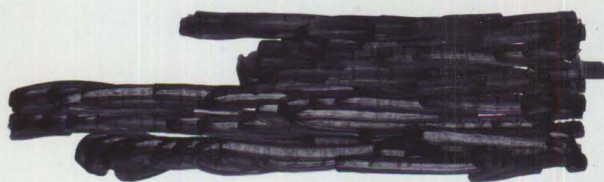
MARCH 1995

Volume 3, Number 3

LAN Storage Management  
& Disaster Recovery

OS/2, UNIX and Oracle —  
An Unlikely Combination?

NetWare and UNIX Interconnectivity



**PC**  
SYSTEMS & SUPPORT  
and **technical**  
SUPPORT  
have merged to form the new  
**technical**  
SUPPORT



## FROM THE PRESIDENT



Dear NaSPA member;

As I write this letter in early February, I have just returned from our quarterly NaSPA board of directors meeting in sunny, warm Dallas, Texas. There's nothing like returning to balmy Wisconsin to re-invigorate my work (and column-writing) ethic.

The NaSPA board of directors this year consists of Emit Hurdelbrink (Denver, Colo.); Mohammad Aziz (Dallas, Texas); James Dreschler (Phoenix, Ariz.); Radi Shourbaji (San Jose, Calif.); and my cold-weather friend, John Suchodolski (New Jersey). We met in Dallas for three days of meetings with NaSPA Membership Marketing Analyst, Tracy Binsfeld, and Rosealee Lee, president of Ardel Inc., a conference planning firm.

Ms. Binsfeld explained the dramatic increase in NaSPA membership since the last board meeting. NaSPA members have responded to the Group Membership offer in great numbers. Up to five members in a data center can sign up for NaSPA memberships at substantial savings. This alleviates the need to "pass along" *Technical Support* throughout the department. If you are interested in signing up a group of your own, check out the cover wrap on the February issue of *Technical Support*, call (414) 423-2420 Ext. 116 or fax (414) 423-2433.

### SPEAKING OF TECHNICAL SUPPORT...

The board reviewed the "new and improved" *Technical Support* and their sentiments have been echoed by many of the comments we have received from you, our members. We hope you share our enthusiasm for the new *Technical Support* and we welcome your comments. Contact Editor Amy Birschbach on NaSCOM via NaSCOM ID EDITOR; on the Internet at editor@nascom.com; or via CompuServe at 70373,1513.

### NASTEC 95 AND 96

I know many of you have been waiting anxiously for news on NaSTEC, NaSPA's premier education conference, and we are pleased to announce that NaSTEC 95, coordinated by Rosealee Lee and James Dreschler, is moving full speed ahead. Scheduled for October/November of this year, NaSTEC 95 will be our 10th Anniversary NaSPA Homecoming and will prove to be the most comprehensive education event of the year! NaSTEC 96, scheduled for March/April 1996, is in the initial planning stages. Stay tuned for the April 1995 issue of *Technical Support* magazine for more information.

### MEMBER SERVICES

The board also discussed NaSPA's member benefits. Current benefits, such as our insurance program and the MCI corporate discount plan were reviewed. Upgrades to NaSCOM were discussed and approved. Watch in the near future for Internet FTP and Telnet services to NaSCOM. V.34 modems and another CD-ROM server were approved and are scheduled for installation this spring.

### PROMISING CHANGES

An 80-page *Technical Support*; increased member benefits; discounts from an expanded list of major education providers; and more opportunities for you, our members, to introduce your colleagues to NaSPA, all point to one thing — we have been listening to you, reading your comments on the surveys and on NaSCOM. We look forward to serving you into the next millennium.

Sincerely,

Scott Sherer

© 1995 Technical Enterprises, Inc.

All Rights Reserved.

Published exclusively for NaSPA, Inc.

### BPA audit applied for December 1994.

4811 S. 76th St., Suite 210

Milwaukee, WI 53220-4362

(414) 423-2420 FAX: (414) 423-2433

MARCH 1995 VOLUME 3, NUMBER 3

## PUBLISHING

### Editor:

Amy B. Birschbach, ext. 123

NaSCOM ID: EDITOR

CompuServe ID: 70373,1513

Internet ID: editor@nascom.com

### Assistant Editor:

Matthew Ringlien, ext. 125

NaSCOM ID: EDIT1

Internet ID: edit1@nascom.com

### Production Coordinator:

Lisa M. Paulin, ext. 124

### Technical Editors:

Eric Allred, Mark Bell,

Craig Collins, Danal Estes, Israel E. Gotay,

Mark Hanna, Howard Hauck,

John E. Johnston, John D. Kinne,

David Kreuter, Leo Langevin, Jim McMaster,

Dwight S. Miller, Steve Pryor,

Stephen L. Samson, Fred Schuff, Al Shing,

Richard B. ViPond, Guy C. Yost

### Editorial Assistant:

Debbie Flatow

## SALES/MARKETING

Jerry Seefeldt

Marketing/Sales Manager, ext. 110

Display Advertising, Card Decks, Reprints

NaSCOM ID: MARKET

Internet ID: market@nascom.com

Cal Hart

East Coast Sales

Display Advertising, Card Decks

(908) 495-6660

Steve Cecil

West Coast Sales

Display Advertising, Card Decks

(415) 595-285

Mike Czarnecki

Sales Representative/

List Manager, ext. 105

Mailing List Sales, PC Merchandise Sales

NaSCOM ID: PCSALES

Internet ID: pcsales@nascom.com

Cover Credit: Jay Bellmore, The Image Bank



# **technical** Support

Supporting Enterprise Networks and Operating Environments

MARCH 1995  
VOLUME 3, NUMBER 3

## FEATURES

### 20 LAN Storage Management & Disaster Recovery

It is the capacity of the LAN to store large amounts of mission-sensitive data that requires the application of storage management and disaster recovery disciplines to those distributed platforms.

By Doug Swords

### 23 OS/2, UNIX and Oracle Part I: An Unlikely Combination?

Managing multiple UNIX systems or systems on different platforms running UNIX can be accomplished through the use of OS/2.

By Robert Simpson

### 34 The Multi-Protocol PC

The combination of graphical interfaces and multiple protocol stacks finally means the user can click on icons to start his applications and neither know or care where the file or application is located or what network is being used.

By Mark Bell

### 38 NetWare and UNIX Interconnectivity Part IV: Printing Across NetWare and UNIX

Cross-platform print sharing is a reality with today's innovative vendor offerings.

By Guy Yost



## COMMUNICATIONS

### 46 Windows-Based World-Wide Web Browsers

With more graphical interface tools becoming available and the rapid proliferation of web pages, the Internet is growing into the richest source of information on the planet, not only for technical information, but for cultural and recreational information as well.

By Al Shing



#### NaSPA Mission Statement:

The mission of NaSPA, Inc., a not-for-profit organization, shall be to serve as the means to enhance the status and promote the advancement of all corporate computing technical professionals; nurture member's technical and managerial knowledge and skills; improve member's professional careers through the sharing and dispersing of technical information; promote the profession as a whole; further the understanding of the profession and foster understanding and respect for individuals within it; develop and improve educational standards; and assist in the continuing development of ethical standards for practitioners in the industry.

The information and articles in this magazine have not been subjected to any formal testing by NaSPA, Inc. or Technical Enterprises, Inc. The implementation, use and/or selection of software, hardware, or procedures presented within this publication and the results obtained from such selection or implementation, is the responsibility of the reader.

Articles and information will be presented as technically correct as possible, to the best knowledge of the author and editors. If the reader intends to make use of any of the information presented in this publication, please verify and test any and all procedures selected. Technical inaccuracies may arise from printing errors, new developments in the industry and/or changes or enhancements to components, either hardware or software.

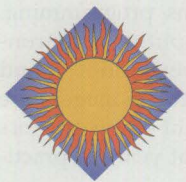
The opinions expressed by the authors who contribute to **NaSPA Technical Support** are their own and do not necessarily reflect the official policy of NaSPA, Inc. Articles may be submitted by members of NaSPA, Inc. The articles should be within the scope of host-based, distributed platforms, network communications and data base, and should be a subject of interest to the members and based on the author's experience. Please call or write for more information. Upon publication, all letters, stories and articles become the property of NaSPA, Inc. and may be distributed to, and used by, all of its members.

NaSPA, Inc. is a not-for-profit, independent corporation and is not owned in whole or in part by any manufacturer of software or hardware. All corporate computing professionals are welcome to join NaSPA, Inc. Membership rates are \$39.95/year (USA), 49.95/year (Canada) and \$59.95/year (all other countries). \$25.00 of your annual dues is allocated to the publication **NaSPA Technical Support** and is non-deductible therefrom.

**NaSPA Technical Support** (ISSN 1079-3135) is published monthly by Technical Enterprises Inc., 4811 S. 76th St., Suite 210, Milwaukee, WI 53220-4362. Second-class postage paid at Milwaukee, WI. POSTMASTER: Send address changes to **NaSPA Technical Support**, 4811 S. 76th St., Suite 210, Milwaukee, WI 53220-4362.

All product names mentioned in this publication are the trademarks/registered trademarks of their respective manufacturers.





## INSIGHTS

### 12 A Comparison of CMS Programming Interfaces Part I: The Program Stack

CMS provides a variety of interfaces, including the CMS program stack, to capture CMS and CP command responses.

By Dave Kreuter

### 16 DB2 V3 Claim and Drain Serialization

The claim and drain process has dramatically improved the performance and availability of DB2 applications by allowing most utilities to run partially or totally parallel with SQL applications.

By Daniel D. Kitay

### 31 Prescription for System Diagnosis

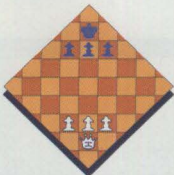
Today's plethora of diagnostic tools can be confusing, but the first step in choosing the appropriate utility is determining your system's problem.

By Paul Buzby

### 55 Taking a Proactive Stance Using Network Management Tools

The key to successful network management is avoidance of network degradation; taking a proactive rather than a reactive approach.

By Kevin Conklin



## STRATEGIES

### 8 Printing Via the JES Spool From CICS

CICS provides a link to satisfy the user's need for hardcopy and eliminates the middle man involved in other JES processes.

By Fred Schuff

### 28 Documenting Your NetWare 3.x Bindery

The MIGRATE.EXE utility updates NetWare bindery versions and aids in converting non-Novell LAN systems to NetWare.

By John E. Johnston

### 50 Imaging Applications Require Strong Networks

Falling prices and a greater availability of storage applications are making electronic storage increasingly cost-effective.

By Henry Wong

## COLUMNS

### 61 MVS Tools & Tricks

Dataset Recovery: Part II  
Fixing the Problems  
By Sam Golob

### 63 VM Toolbox

Console Council  
By John D. Kinne

### 65 VSE Tools & Techniques

Displaying VSAM Lock Resources  
By Mark Hanna

### 67 Storage Management

Setting Allocation Defaults  
With ALLOCxx  
By Steve Pryor

### 69 Opening Windows

Touring Microsoft's  
World-Wide Web Server  
By Al Shing

### 70 OS/2 Insights

Invoking NetWare VLMs  
Under OS/2  
By John E. Johnston

### 71 Open Systems Solutions

Reflections on Electronic Mail  
By Harold Hauck

### 74 Security Strategies

Securing Large Networks  
By Eric Allred

### 76 On a Personal Note

Service Provider  
or Consultant?  
By Donald R. Fowler  
and Michael K. Sutton

## DEPARTMENTS

### 3 From the Editor

### 6 NaSPA News

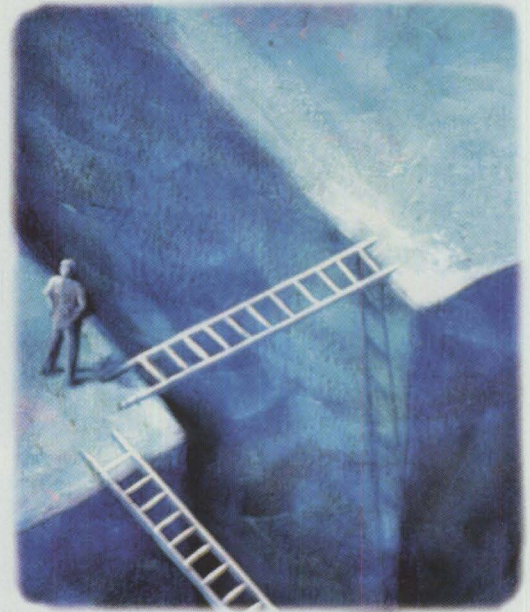
### 57 BBS Buzz

### 77 Product Profiles



# OS/2, UNIX and Oracle Part I: An Unlikely Combination?

*Managing multiple UNIX systems or systems on different platforms running UNIX can be accomplished through the use of OS/2.*



**T**his is the first of a four-part series on using the OS/2 operating system to manage Unix systems and Oracle databases. This article presents an approach for managing Unix systems with OS/2. Part II shows how to use Unix shell scripts to perform functions which can be valuable in an Oracle environment. Part III expands this approach to managing the Oracle DBMS using scripts written for one of the Oracle utilities, SQL\*Plus. The concluding article will present a simple C language program which cleans up the output and allows generating printed reports using the approach presented in the first three parts.

If you are responsible for managing a single UNIX system and consider yourself a UNIX expert, then you probably will not agree with the approach described in this series of articles. On the other hand, if you're a mainframe systems programmer or PC technical coordinator who has recently been given the responsibility of managing systems running UNIX, are responsible for managing multiple UNIX systems, especially on different platforms, or are already familiar with OS/2 and would like to take advantage of it in managing your UNIX systems, you might be interested in the approach presented here.

## ADVANTAGES

Some advantages of using OS/2 to manage UNIX systems, as opposed to other PC-based operating systems, include:

■ **Long file names** — DOS, Windows and the NetWare Client for OS/2 (at least for

NetWare 3.X servers) limit file names to the "8.3" format. With the long file names feature of OS/2's High Performance File System (HPFS), files can be moved between UNIX and the PC without having to rename the ones which do not conform to the DOS naming conventions.

■ **Familiar commands** — using TCP/IP's network client support, just about any DOS or OS/2 user should have no trouble navigating UNIX, using familiar commands such as "DIR" and "TYPE" under OS/2.

■ **The OS/2 editors** — files can be created and modified using the familiar OS/2 editors rather than using the UNIX "vi" editor.

## CAVEATS

■ **HPFS vs. FAT** — in order to use long file names in parameters for commands entered at the OS/2 command prompt, the hard drives specified by those parameters must be formatted as HPFS. Drives formatted with the FAT (File Allocation Table) file system support long file names only through the Workplace Shell.

■ **Termination** — OS/2 and DOS terminate each line of a text file with a carriage return and line feed and add an end-of-file marker while UNIX terminates lines with just a carriage return and omits the end-of-file marker. Some utilities handle these differences automatically.

## TCP/IP FOR OS/2

TCP/IP is the standard protocol for communicating with UNIX systems. A TCP/IP communications protocol stack

is needed to access UNIX systems from other operating systems. A number of vendors provide TCP/IP products for OS/2. The applications normally included with TCP/IP provide a nice variety of useful functions. The flexibility of these applications and the compatibility of OS/2 with UNIX are an almost ideal combination.

IBM's TCP/IP for OS/2 product line consists of a number of related and overlapping products, called "kits." To execute the examples provided with this series of articles, the Base Kit (65G122) and the NFS Kit (65G1255) will be needed. The Base Kit contains the TCP/IP protocols and a number of applications such as Telnet, a terminal emulator, and FTP, the File Transfer Protocol client program.

The NFS Kit provides the networking function. The NFS client allows an OS/2 system to access files on a UNIX system using a network drive letter under OS/2. With the NFS server an OS/2 system can "export" a directory structure to allow a

**Figure 1: Sample FTP Session**

```
? ftp yourHostname
Username: yourUsername
Password: yourPassword
FTP> cd yourDirectory
FTP> pwd
/usr/yourUsername/yourDirectory
FTP> get remotefile localfile
765 bytes transferred
FTP> put localfile remotefile
1234 bytes transferred
FTP> quit
```



**Figure 2: Using NFS Under TCP/IP for OS/2**

```
[D:\] nfsstart
[D:\] showexp
export list for yourHostname:
/uttl          everyone
[D:\] mount -l yourUsername -p yourPassword u: yourHostname:/uttl
NFS Drive 'u:' was attached successfully.
[D:\] dir u:
```

```
The volume label in drive U is NFS.
The Volume Serial Number is 001E:0000
Directory of U:\
```

12-14-94	9:25a	<DIR>	0	.
12-12-94	10:16p	<DIR>	0	..
12-08-94	8:20p	<DIR>	0	dba
12-08-94	1:14p	<DIR>	0	doc
11-10-94	7:55p	<DIR>	0	bin
12-07-94	3:15p	<DIR>	0	cmd
6 file(s)		3632 bytes used		
		689103872 bytes free		

**Figure 3: Directory Structure for the Examples**

```
util - main repository directory
-> bin - subdirectory for UNIX executable files
-> cmd - subdirectory for OS/2 REXX command procedures and
      other executable OS/2 files
-> dba - subdirectory for Oracle SQL*Plus scripts used in Part II
```

UNIX system to access its files. There is also a Total Kit which includes both the Base Kit and the NFS Kit, as well as software which can run X Windows applications and transmit TCP/IP data over an SNA network.

Although OS/2 Warp 3's BonusPak includes some TCP/IP function, it does not support a LAN connection. If you need to access UNIX systems on a LAN, get the TCP/IP for OS/2 product kits or the OS/2 Warp 3 LAN Client package.

Telnet is the TCP/IP component which allows access to UNIX systems as if connected via a terminal. While UNIX systems can be managed in the traditional manner equally well from any environment providing Telnet capability, the purpose of these articles is to show one way to take advantage of the other TCP/IP functions used in conjunction with OS/2.

### FILE TRANSFER — FTP

FTP is the basic file transfer capability which comes with or can be added to most versions of TCP/IP. A typical file transfer session might look like the sample in Figure 1. The change directory "cd", print working directory "pwd", and "dir" commands apply to the directory on the remote system. The second file name in the "get" and "put" commands is necessary only if the file is to be renamed. Files must be renamed when transferring files with names not supported on the target system; for example, when files that do not fit within the "8.3" naming convention are being transferred to a FAT drive on an OS/2 system. To "get" a file, you must have "read" permission on the file. To "put" files you must have "write" permission on the directory and any files to be overlaid. The owner of the file can use the UNIX "chmod" command to change the permissions (enter "man chmod" for more information).

An FTPPM utility in IBM's TCP/IP for OS/2 provides the same function as the command line FTP program, but through a friendlier interface.

### NETWORK FILE ACCESS — NFS

NFS, the Network File System, is another way to access files on remote UNIX systems. A directory can be "mounted" on another system to make it appear as part of its own file system. However, before you can remotely access files on a UNIX system, the NFS "daemon" (nfsd) must be running and part of the UNIX file system must be "exported" by adding a directory name to the "/etc/exports" file and issuing the "exportfs -a" command. Figure 2 shows how NFS might be used to access files which have been exported from a UNIX system. After testing it at the OS/2 command prompt, the complete "mount" command can be placed in the OS/2 file "\tcpip\etc\fstab" to be executed automatically each time NFS is started.

A UNIX directory mounted on an OS/2 system appears as an additional drive letter. Normal OS/2 commands such as DIR, COPY and EPM can be used to manipulate files. Again, you must have the appropriate UNIX permissions to access the files. An OS/2 directory can be exported and mounted on a UNIX system, giving users on the UNIX system access to the OS/2 files.

The difference between OS/2 and UNIX file formats was mentioned earlier. Two utilities are supplied with TCP/IP for OS/2 — "OS22UNIX" and "UNIX2OS2" — to convert files from one format to the other, when necessary. These are "filters" which can be used by redirecting the standard input and output files.

### MANAGING FILES ACROSS MULTIPLE SYSTEMS

When managing a single set of files across multiple systems, it's a good idea to choose one of the systems as the primary repository for the files. The choice may depend on a number of factors, such as the number of systems being managed, which systems have backup procedures in place, system availability, ability to access remote systems, ease of recovery, who needs access to the files, your preference for operating systems, and the methods available for modifying the files. The repository for the files described in this series can be an OS/2 system, a network file server or a UNIX system.

**If you've recently been given the responsibility of managing systems running UNIX and are familiar with OS/2 and would like to take advantage of it in managing your UNIX systems, you might be interested in the approach presented here.**

Although the UNIX files could be stored on a UNIX system and the OS/2 files could be stored on an OS/2 system or file server, this results in different types of files being stored in different locations. Keeping all of the files in one location allows the same programs and procedures to be used for updating and distributing the various types of files.

### REPOSITORY DIRECTORY STRUCTURE

It's helpful to keep different types of files separate; store them in different subdirectories. Figure 3 shows the subdirectory structure used for the examples throughout this series.



The same directory structure can be used on both UNIX and OS/2 systems.

Figure 4 shows which files need to be accessible by the various OS/2 and UNIX systems involved. Note that there is one area where OS/2 is incompatible with UNIX, due to a carryover from DOS. In UNIX, a forward slash, "/", separates directory names in a file path, while in DOS and OS/2, a backslash character "\" is used. Providing access to the files can be accomplished by distributing them to each system, or by sharing the repository using NFS.

#### DISTRIBUTING FILES

It may be necessary to store some files in multiple locations. Using a system which does not provide a NFS server capability as the repository requires the files to be duplicated on the other UNIX systems which do not hold the repository.

If your repository is on OS/2 or a file server, the files will need to be copied from the repository to each of the UNIX systems. You might also decide to store files in multiple locations for performance or other reasons. Once the same files are stored in multiple locations, a method of updating them is needed. Either the TCP/IP File Transfer Protocol client programs, FTP and FTPPM, or the network file system, NFS, can be used.

The files could be transferred to each system as needed or a procedure could be set up to distribute modified files, keeping the files on all of the systems together. Figure 5 shows a REXX command procedure, DISTRIB.COM, which can be used to distribute files from an OS/2-based repository to multiple UNIX systems using either FTP or NFS. A number of constants near the beginning of the file will need to be modified for your specific environment.

To use it, change to the main repository directory ("util" in Figure 3) and enter the command "DISTRIB BIN" to distribute any modified UNIX executable files in the "bin" subdirectory. If distribution to one or more hosts fails, it can be retried by entering "TO yourHostname". If there is a problem, it may be helpful to look at the statements generated in the "TO.COM" file.

#### EDITING UNIX FILES UNDER OS/2

There are a number of ways UNIX files can be created and modified under OS/2, depending on where the repository is located. If the files are stored on an OS/2 system, or are accessible to OS/2 using NFS, the OS/2 System Editor or the Enhanced Editor (EPM)

Figure 4: Files Needed (Depending on Location of Repository)

OS/2 or Network Repository			
Type of Files	Repository	Other OS/2 systems	UNIX systems
OS/2 commands			
common	\util\cmd	u:\cmd	(not needed)
user-specific	N/A	\util\cmd	(not needed)
UNIX executable			
common	\util\bin	(optional *)	/util/bin
user-specific	N/A	(not needed)	/usr/username/bin
(samples of the user-specific scripts can be stored in \util\bin)			
Oracle scripts	\util\dba	(optional *)	/util/dba
* For a network-based repository, the same files can be shared by multiple OS/2 users, allowing them to be modified more easily.			
UNIX Repository			
Type of Files	UNIX Repository	Other UNIX systems	OS/2 Systems
OS/2 commands			
common	/util/cmd	(not needed)	u:\cmd
user-specific	N/A	(not needed)	\util\cmd
UNIX executable			
common	/util/bin	/util/bin	(not needed)
user-specific	/usr/username/bin	/usr/username/bin	(not needed)
Oracle scripts	/util/dba	(optional)	(not needed)

can be used to create and modify the files. This is where the difference between the file formats needs to be considered.

The files can be transferred from UNIX to OS/2 using FTP, edited normally, then transferred back. Figure 6 shows a REXX command procedure which automates

### MULTI-PLATFORM DATA COMPRESSION

## COMPRESS YOUR COSTS AND SAVE!

DOS

MVS

Windows

AS/400

Mac

UNIX

ASi is the pioneer of open systems/multi-platform data compression. We are dedicated to providing cost-effective solutions that can save your company money. Powerful members of our data compression line-up include such well known names as PKZip® and WinZip®. We offer compression in stand alone and developer's versions. Call ASi today and compress your costs!

**Try us out today!**

Call the NaSCOM  
24hr. BBS at:  
414-28-NASPA.  
Download "I" from  
the "Must Have"  
section.

- Cut data transmission costs and time by 2X10 times more!
- Share data quickly and efficiently between different systems in the industry's most popular format!
- Error eliminating 32 Bit CRC

Call us today at 513-885-2031  
for more details and to arrange  
your FREE 30 day evaluation.\*

\*Nominal shipping and handling charges.



#### COMPATIBLE DATA COMPRESSION

ASCENT SOLUTIONS Inc. (ASi)  
10460 Miamisburg-Springboro Pike, Miamisburg, OH 45342  
(513) 885-3032 FAX • (513) 885-2033 Data



**Figure 5: DISTRIB.COMD - Distributing Files With FTP or NFS**

```
/* REXX command procedure to distribute files from OS/2 to UNIX */
parse arg subdirs discard

/* Change the following constants as needed for your environment */

method = NFS

dest.1 = 'hostname1 /util yourUsername yourPassword'
dest.2 = 'hostname2 /util yourUsername yourPassword'
dest.3 = 'hostname3 /util yourUsername yourPassword'

ftptemp = 'ftp.tmp'
nfsdrive = 'v:'

ofile = 'to.cmd'
lstfile = 'distrib.lst'

/* Beginning of procedure */

if subdirs = '' then subdirs = '.'

'@del' lstfile
'@del' ofile
'@dir /aa /b /l' subdirs'\*'lstfile' 2>nul'

call lineout ofile, '/* REXX */'
call lineout ofile, 'parse arg system path username password discard'

count = 0

if method = FTP then do
  call lineout ofile, "'echo user' username password '>'ftptemp'"
  call lineout ofile, "'echo cd' path'/'subdirs' >>'ftptemp'"
  do while ( lines(lstfile) )
    iline = linein(lstfile)
    if iline \= lstfile & iline \= '' then do
      oline = "'echo put' iline" >>'ftptemp'"
      call lineout ofile, oline
      count = count + 1
    end /* Do */
  end /* do */
  call lineout ofile, "'echo quit >>'ftptemp'"
  call lineout ofile, "'ftp -n' system '<'ftptemp'"
end /* Do */
else if method = NFS then do
  call lineout ofile, "'mount -l'username '-p'password '"nfsdrive'"
  system':path' do while ( lines(lstfile) )
    iline = linein(lstfile)
    if iline \= lstfile & iline \= '' then do
      oline = "'os22unix <'iline">'nfsdrive || subdirs'\iline'"
      call lineout ofile, oline
      count = count + 1
    end /* Do */
  end /* do */
  call lineout ofile, "'umount '"nfsdrive'"
end /* Do */
else do
  say "'method' in DISTRIB.COMD must be FTP or NFS"
  exit
end /* Do */

call stream ofile, 'c', 'close'

if count = 0 then do
  say "No files to distribute"
  exit
end /* Do */

if method = NFS then do
  'umount' nfsdrive
end /* Do */

i = 1
do while dest.i \= 'DEST.i'
  'call to.cmd' dest.i
  i = i + 1
end /* do */

'@attrib -a' subdirs'\*'

say 'Enter "TO system path yourUsername yourPassword" to'
say 'retry distribution to any systems which failed above'

exit
```

the file transfer before and after invoking the editor. To use the command type:

```
start /c ftpedit /path file
```

at the OS/2 command prompt, replacing `"/path"` with the UNIX path name and `"file"` with the name of the file to be edited.

While FTP automatically converts files between OS/2 format and UNIX format, NFS does not. If you use COPY or XCOPY to copy files created with an OS/2 editor to a UNIX system via NFS, or create the files directly, with EPM for example, the files will be in OS/2 format.

The "OS22UNIX" utility supplied with TCP/IP for OS/2 can be used to remove the carriage returns and end-of-file marker, resulting in files in UNIX format. There is also a "UNIX2OS2" utility to convert UNIX files back to OS/2 format, but since the EPM editor will read and properly interpret files in UNIX format, you can avoid using it when modifying UNIX files under OS/2. Note however, that some UNIX files, such as `".login"` and shell scripts, should not be edited this way since these files may generate errors when saved in OS/2 format rather than the proper UNIX format. Fortunately, Oracle's SQL\*Plus utility, used in Part III, has no problem handling files in either format.

An alternative method for altering the UNIX files is to use Telnet to access the UNIX system, and a UNIX editor, such as `"vi"`, to modify the files. Files which are created in this manner will be stored in UNIX format. If you use the `"vi"` editor to modify a file which was created with an OS/2 or DOS editor and transferred to UNIX with NFS without the "OS22UNIX" filter, the carriage return characters will show up as `"^M"` and the end-of-file marker will show up as `"^Z"`. In `"vi"`, these extraneous characters can be removed by typing the `"x"` key; each press of the `"x"` key will remove both the `"^M"` and the character following it, since these combinations represent a single non-displayable character.

### **ACCESS TO FILES USING NFS**

Using NFS to share the files among multiple systems avoids the need to have the same files distributed to multiple locations. Access to the files is provided by exporting the repository and mounting it on the other systems in the network. The repository in the examples can be exported by adding a line with `"/util"` to the UNIX `"/etc/exports"` file and entering the command `"exportfs -a"`. After



mounting the directory as shown in Figure 2, files can be updated directly, using an editor for example, without having to perform an additional step to distribute the updated files to other systems.

For simplicity, the remainder of this article will assume that the files are stored in only one location but can be accessed from any OS/2 or UNIX system using NFS. The examples assume that this repository has been mounted so it can be accessed from OS/2 systems using the drive letter "u:".

### MODIFYING CONFIG.SYS

To allow the REXX command files in the "cmd" subdirectories on the repository and on your hard disk to be executed without specifying their path, add these directories to the "SET PATH" statement in the OS/2 CONFIG.SYS file. Add "d:\UTIL\CMD;u:\cmd;" to the end of the "SET PATH" statement:

```
SET PATH=d:\OS2;d:\OS2\SYSTEM;
d:\OS2\INSTALL;d:\d:\OS2\MDOS;
d:\OS2\APPS;d:\TCP\BIN;
d:\UTIL\CMD;u:\cmd;
```

where "d:" is the drive letter for your hard disk, and "u:" is the drive letter for the repository. You must either reboot or issue the command:

```
SET PATH=%PATH%;d:\UTIL\CMD;u:\cmd;
```

in each opened OS/2 Window to make the change take effect.

### EXECUTING UNIX COMMANDS

Now that we know how to create the files and where to store them, how do we tie the UNIX and OS/2 environments together? Figure 7 shows a short user-specific REXX command file "cmdu.cmd" (the "u" stands for UNIX and distinguishes the command name from OS/2's CMD.EXE) which executes a single UNIX command using the "rexe" function of TCP/IP for OS/2. Since the command contains your own username and password, it should be stored in "\util\cmd" subdirectory on your hard disk, "d:", not in the repository. The UNIX command to be executed is simply passed as parameters to the command, for example:

```
cmdu ls -l -t
```

will list the contents of your home directory in reverse chronological order.

Figure 6: FTPEDIT.CMD - Editing a UNIX File via FTP

```
/* REXX command procedure to edit UNIX files with EPM via FTP */
parse arg path file

if file = '' then do
    say "Usage: ftpedit <path> <file>"
    exit
end

/* Do */

/* Change the following constants as needed for your environment */

username = 'yourUsername'
password = 'yourPassword'
system = yourHostname
ftptemp = 'ftp.tmp'

/* Put temp files in directory indicated by the "TMP" environment variable */

tempdir = value("TMP",,"OS2ENVIRONMENT")
if tempdir \= '' then tempdir = tempdir || '\ '

ftptemp = tempdir || ftptemp
epmfile = tempdir || file

/* Beginning of procedure */

'@echo off'
'echo user' username password '>'ftptemp
'echo cd' path '>>'ftptemp
'echo get' file epmfile '>>'ftptemp
'echo !epm /m' epmfile '>>'ftptemp
'echo put' epmfile file '>>'ftptemp
'echo quit >>'ftptemp
'ftp -n' system '<'ftptemp

exit
```

Figure 7: d:\util\cmd\cmdu.cmd

```
/* CMDU.CMD */
parse arg args
"@rexe yourHostname -l yourUsername -p yourPassword" args
```

The command which is passed as a parameter to "cmdu" could be just about any UNIX command or a shell script containing a number of commands. When possible, use a modular approach, separating the general-purpose functions from application-specific functions.

### SECURITY CONSIDERATIONS

One thing this series of articles does not address is security. The REXX command files presented in this series contain UNIX and Oracle user names and passwords. The user names and passwords could be omitted from the command files presented in this article and Part II. These items would then have to be supplied from the keyboard each time the command files are executed. Note however, that it is not possible to redirect the output of the scripts to, for example, a printer without also redirecting the prompts for username and password, which will cause problems when we get to Part IV.

The second part of this series will provide several examples of OS/2

REXX command procedures and UNIX shell scripts which may be helpful if you are running an Oracle database on UNIX. **ts**

*Was this article of value to you? If so, please circle Reader Response Card No. 24.*



*NaSPA member Robert Simpson has more than 16 years computing experience, specializing in systems software support. He is experienced in installing and supporting OS/2 and related communications software, as well as data base and communications software on the MVS/ESA platform. He can be reached via CompuServe ID 71520,737 or Internet address 71520.737@compuserve.com.*



# Invoking NetWare VLMs Under OS/2

BY JOHN E. JOHNSTON

**T**he latest version of NetWare includes a great utility known as NetWare User Tools. This Windows-based utility provides a graphical interface from which you can map drives and connect your LPT ports to NetWare print queues. You can also configure the NetWare User Tools utility to invoke user-defined functions. This utility is great for networked Windows users as it allows them to perform many functions they could not easily do before.

The only problem OS/2 users will encounter is that this utility requires loading NetWare Virtual Load Modules (VLMs). Loading these VLMs in a DOS session under OS/2 is no simple task, but with a bit of persistence it can be done. At the time of this writing, the instructions provided by Novell for implementing the VLMs under an OS/2 DOS session did not work.

## DOWNLOADING THE LATEST OS/2 REQUESTER

Before you can implement the VLMs under OS/2, you must first download and install the latest NetWare OS/2 Requester. To download the new Requester, login to CompuServe and enter the NOVFILES forum. From this forum, select "Client Kits," then "OS/2 Client Kit," then select "NetWare Client v2.11 for OS/2." You must download the following files from this area:

WSOS21.EXE;  
WSOS22.EXE;  
WSOS23.EXE;  
WSDRV1.EXE;  
OS2UT1.EXE;  
OS2DC1.EXE; and,  
OS2VMB.EXE.

Expect to spend about an hour downloading these files using a 9600 BPS modem.

Installing the new requester is relatively simple; there's a documentation file included in the WSOS21.EXE file that

explains the entire process. If you follow these instructions you should have very few problems. I have found that it's always safer to remove your current requester before installing a newer version. To do this, remove all lines in your CONFIG.SYS file that load programs from the NetWare directory. Also, remove any references to the NETWARE directory from your PATH, LIBPATH and DPATH statements.

## Loading these VLMs in a DOS session under OS/2 is no simple task, but with a bit of persistence it can be done.

Once these modifications have been performed, restart your system and delete all entries from the NETWARE directory and then remove the directory. As usual, it's important to make a backup of the NETWARE directory, your CONFIG.SYS, NET.CFG and PROTOCOL.INI (if required). Follow the directions in the new requester's documentation file to install the new version. When finished, re-boot your system and test the software.

## CREATING THE VLM BOOT IMAGE

From a real DOS machine (not an OS/2 DOS session) format a diskette using the following command:

"FORMAT A: /S".

Place a scaled-down CONFIG.SYS and AUTOEXEC.BAT file on this boot diskette and label it VLMBOOT using the LABEL command. Go back to your OS/2 system with the new requester loaded and double click on the Novell icon, then click on the Install icon from within the

Novell folder. After the installation utility initializes, click on the Configuration pull-down item and select "VLM Boot Setup." A dialog box will be displayed from which you select the diskette drive that will contain the VLM boot diskette.

Insert the diskette and make sure this dialog box contains the proper drive letter for your system, then click on OK. Another dialog box will be displayed asking you to place the diskette labeled VLMBOOT in your diskette drive. Click on OK. The next dialog box will tell you that your AUTOEXEC.BAT file has been updated. Click on OK and another dialog box will appear with the following prompt: "Create NetWare VLM Boot Image File." Click on the box to the left of this prompt, making sure the proper diskette drive letter is in the lower box, then click on OK. When the utility completes a VLM Boot icon will be placed on your OS/2 Desktop.

## INVOKING NETWARE USER TOOLS

To test the VLM boot process, double-click on the VLM Boot icon. A DOS session will be started and you should see the VLMs load. Type WIN to start WIN-OS2. If you look around you will see that you cannot find the NetWare User Tools icon within your WIN-OS2 session. You must obtain the program NWUSER.EXE from a DOS machine that has this utility loaded on it. (Note: NWUSER.EXE is installed using the NetWare WSWIN\_1 diskette and is explained in the *NetWare Workstation for DOS and Windows* manual.)

Once you have your hands on NWUSER.EXE, copy it to your C:\OS2\MDOS\WINOS2 directory (or another directory if desired). Create a new Windows Program Group and Item to invoke NWUSER.EXE. If all goes well, the utility should start right up.

## LIMITATIONS OF NWUSER UNDER OS/2

Most of you are aware that you can have both Global and Private NetWare sessions under OS/2 DOS sessions.




# Reflections on Electronic Mail

BY HAROLD HAUCK

**The problem with NWUSER is that it must run in a Private session which can severely limit the utility's usability in shops where Global sessions are the norm.**

## TEST TROUBLESOME APPS

The creation of the VLM Boot image and icon can be used for other functions as well as invoking the NetWare User Tools. You can use the boot image to test applications that are "acting up" on your DOS session. This can help to isolate problems with the OS/2 Requester. Even if you do not need the VLM Boot functionality, you should still implement the new OS/2 NetWare Requester. This latest version corrects many problems and appears to be very stable.

*If you have any questions, comments or ideas for future topics for this column, feel free to contact me via NaSCOM at Johnjohe or CompuServe at 73473,2146. *

*Was column of value to you? If so, please circle Reader Response Card No. 38.*



**NaSPA member John E. Johnston is manager of technical support and communications for a major hospital in Pennsylvania. He designs and maintains cross-platform local and wide area**

**networks utilizing NetWare, OS/2, DOS and Windows.**

**R**ecently I began thinking about our expanding communications capabilities, and started to reflect upon one of the technologies which has become almost as pervasive as the computers on which it is based.

Email. Just about everybody with a computer or desktop terminal is using this easy and very powerful tool to communicate with their business associates, customers and friends. In the not too distant past, only people who used mainframes and mini-computers were able to correspond electronically, and even then the communication was limited, in most cases, to the organization's computer network. For many years, while employed by IBM, I could only exchange email with other IBM employees. Of course that situation changed in the 1980s when IBM established a mail gateway to the Internet. Suddenly, I and all my IBM associates could send and receive email with anyone with Internet access, anywhere in the world. My email horizon expanded significantly, but communication was still limited to people working in organizations with computer access to the Internet.

Today, the number of people capable of using email is expanding well beyond those individuals who work for organizations with computer networks. If you have a PC in your office or home, chances are you have modem access to an email provider such as CompuServe, Prodigy, America Online, MCI, Sprint, AT&T, Advantis, etc. All of these commercial network providers offer email services to their subscribers. The thing that makes email so powerful today is not that these providers and most of corporate America offer email capability to their employees and clients, but rather, the almost universal connection of organizations and commercial network providers to the Internet.

The beauty of the Internet is that it allows people using a wide variety of mail systems to exchange mail. An IBM PROFS

user can send mail to a UNIX, cmail, CompuServe or MCI user as easily as sending to another PROFS user. And of course, users of those email systems can exchange messages between each other and our hypothetical PROFS user with equal ease. The Internet has provided us with universal email interoperability.

## WHAT'S BEHIND THE INTERNET?

What is it about the Internet that has so drastically affected the scope of email systems? The answer, of course, is the TCP/IP protocol suite upon which the Internet is based. More specifically, the Simple Mail Transfer Protocol (SMTP) and the MAIL protocols are the catalysts, or in more technical terms, the defacto standards which all modern mail systems implement to ensure electronic message interoperability.

SMTP and MAIL are two of the protocols designated as standards in 1990 by the Internet Activities Board (IAB). These protocols are described by Request For Comment documents: SMTP is RFC821 and MAIL is RFC822. If you are interested in the technical details of email systems, these documents may be obtained by anonymous ftp from nic.ddn.mil.

The strength of these protocols is in their simplicity. Operating on the client/server model, SMTP specifies the exact format of messages a client process on one machine must use to transfer mail to a server process on another machine. It specifies the commands and sequence of events required for computers to exchange mail. However, the beauty of the SMTP protocol lies more in what is not addressed than what is specified. For example, it does not talk about how the mail system obtains messages from the user, or how mail is presented to users. How mail messages are stored or how frequently computers must communicate to exchange mail are all issues left to the mail system developers. These items are not essential for the exchange of mail, but they